

LOCAL SEARCH

MultiKnapsack with MinMaxType Constraints

Nguyễn Đình Thắng, Đỗ Văn Tuấn

2019 - 05

- 1 Mô tả bài toán MultiKnapsack
- 2 Phương pháp 1
 - Mô hình toán học bài toán MultiKnapsack
 - Ý tưởng thuật toán
- 3 Phương pháp 2
 - Mô hình toán học bài toán MultiKnapsack
 - Ý tưởng thuật toán
- 4 Kết quả thử nghiệm

1 Mô tả bài toán MultiKnapsack

2 Phương pháp 1

- Mô hình toán học bài toán MultiKnapsack
- Ý tưởng thuật toán

3 Phương pháp 2

- Mô hình toán học bài toán MultiKnapsack
- Ý tưởng thuật toán

4 Kết quả thử nghiệm

Mô tả bài toán MultiKnapsack

Cần sắp xếp N items vào M bins, trong đó:

- Item i ($i = 1, \dots, N$) có:
 - $w[i]$: trọng số 1,
 - $p[i]$: trọng số 2,
 - $t[i]$: thể loại của item i ,
 - $r[i]$: lớp của item i ,
 - $D[i]$: tập các bin mà item có thể xếp vào
- Bin b ($b = 1, \dots, M$) có:
 - $W[b]$: tải tối đa cho trọng số 1,
 - $LW[b]$: tải tối thiểu cho trọng số 1,
 - $P[b]$: tải tối đa cho trọng số 2,
 - $T[b]$: số lượng thể loại tối đa trong bin b ,
 - $R[b]$: số lượng lớp tối đa trong bin b

Mô tả bài toán MultiKnapsack

Các ràng buộc của bài toán:

- Với mỗi bin b :
 - C1: Bin b có thể được sử dụng hoặc không. Nếu bin b được sử dụng thì tổng trọng số 1 của các item được xếp bin b phải lớn hơn hoặc bằng $LW[b]$ và nhỏ hơn hoặc bằng $W[b]$
 - C2: Tổng trọng 2 của các item được xếp bin b phải nhỏ hơn hoặc bằng $P[b]$
 - C3: Tổng số thể loại của các item được xếp vào b phải nhỏ hơn hoặc bằng $T[b]$
 - C4: Tổng số lớp của các item được xếp vào b phải nhỏ hơn hoặc bằng $R[b]$
- Với mỗi item i :
 - C5: item i chỉ được xếp vào một bin duy nhất
 - C6: item i chỉ được xếp vào một trong các bin thuộc tập $D[i]$

- 1 Mô tả bài toán MultiKnapsack
- 2 Phương pháp 1
 - Mô hình toán học bài toán MultiKnapsack
 - Ý tưởng thuật toán
- 3 Phương pháp 2
 - Mô hình toán học bài toán MultiKnapsack
 - Ý tưởng thuật toán
- 4 Kết quả thử nghiệm

Mô hình toán học theo phương pháp 1

Mô hình hóa

- Biến:

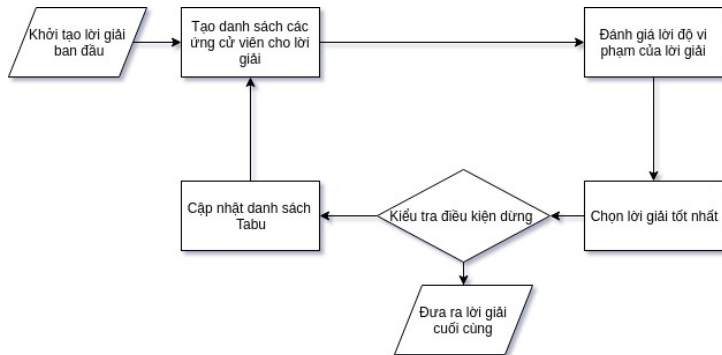
- $X[i, b] = 1$: item i được xếp vào bin b , $X[i, b] = 0$: item i không được xếp vào bin b
- $Y[i, b] = 1$: bin b có thể loại i
- $Z[t, b] = 1$: bin b có lớp i

- Ràng buộc:

- $\sum_{i=1}^N w[i] * X[i, b] = 0 \vee LW[b] \leq \sum_{i=1}^N w[i] * X[i, b] \leq W[b],$
 $b = 1, \dots, M$
- $\sum_{i=1}^N p[i] * X[i, b] \leq P[b], b = 1, \dots, M$
- $X[i, b] \leq Y[t[i], b], i = 1, \dots, N, b = 1, \dots, M$
- $X[i, b] \leq Z[r[i], b], i = 1, \dots, N, b = 1, \dots, M$
- $Y[i, b] \leq T[b], b = 1, \dots, M$
- $Z[i, b] \leq R[b], b = 1, \dots, M$
- $\sum_{b=1}^M X[i][b] = 1, i = 1, \dots, N$
- $\sum_b^{D[i]} X[i][b] = 1, i = 1, \dots, N$

Ý tưởng thuật toán

Sử dụng Tabu Search: Bắt đầu từ một lời giải khởi tạo ban đầu, lặp tìm kiếm các lời giải láng giềng có vi phạm nhỏ nhất với điều kiện lời giải đó nằm ngoài danh sách các đỉnh đã duyệt qua gần nhất. Cài đặt bằng thư viện OpenCBLIS.



Hình: Sơ đồ Tabu Search

- 1 Mô tả bài toán MultiKnapsack
- 2 Phương pháp 1
 - Mô hình toán học bài toán MultiKnapsack
 - Ý tưởng thuật toán
- 3 Phương pháp 2
 - Mô hình toán học bài toán MultiKnapsack
 - Ý tưởng thuật toán
- 4 Kết quả thử nghiệm

Mô hình toán học theo phương pháp 2

Mô hình toán

- Biến:
 - $X[i]$ là bin mà item i được xếp vào
- Ràng buộc:
 - $\sum_{i=1}^N w[i] * (X[i] == b) = 0 \vee LW[b] \leq \sum_{i=1}^N w[i] * (X[i] == b) \leq W[b], b = 1, \dots, M$
 - $\sum_{i=1}^N p[i] * (X[i] == b) \leq P[b], b = 1, \dots, M$
 - $COUNT_DISTINCT_{i=1}^N t[i] * (X[i] == b) \leq T[b], b = 1, \dots, M$
 - $COUNT_DISTINCT_{i=1}^N r[i] * (X[i] == b) \leq R[b], b = 1, \dots, M$
 - $X[i] \in D[i], i = 1, \dots, N$

Sử dụng tabu search giống với phương pháp 1. Cài đặt bằng thư viện OpenCBLS.

Bổ sung thêm 2 class:

- **ConditionCount** (VarIntLS[] X, int[] w, int val) : Đếm số lượng các phần tử khác nhau của một tập các phần tử w[i] với điều kiện i thỏa mãn $X[i] = \text{val}$.
- **IsIn** (VarIntLS x, int[] D) : Ràng buộc x thuộc tập D, nếu x thuộc D violation = 0, nếu x không thuộc D thì violation = 1.

- 1 Mô tả bài toán MultiKnapsack
- 2 Phương pháp 1
 - Mô hình toán học bài toán MultiKnapsack
 - Ý tưởng thuật toán
- 3 Phương pháp 2
 - Mô hình toán học bài toán MultiKnapsack
 - Ý tưởng thuật toán
- 4 Kết quả thử nghiệm

Kết quả thử nghiệm

Các tập dữ liệu

File	Số Item	Số Bin	Violation	
			phương pháp 1	Phương pháp 2
test (trong đề bài)	16	3	0	0
Input	100	1846	thời gian chạy quá lâu	thời gian chạy quá lâu
Input-1000	1000	1846	tràn bộ nhớ	tràn bộ nhớ
Input-3000	3000	1846	tràn bộ nhớ	tràn bộ nhớ